ONNX Community Day 2023 - June 28

ONNX Community Meetup
Wednesday, June 28 | 9am-1pm PST | NVIDIA, Santa Clara, CA | RSVP

All the videos could be found at our youtube channel https://www.youtube.com/onnxai . The playlist for the meetup is here: https://www.youtube.com/watch?v=tWk1sG6vYQI&list=PLNqLlwTfo322wVVTsH_ocKtSj8FEEnVpJ&pp=iAQB . The videos will also be added here at confluence for the people who are not able to access youtube.

The detailed agenda for this event is listed below:

9:00am: Introduction (15 mins)	
Welcome	**Kari Briski**, NVIDIA **Mayank Kaushik**, NVIDIA
ONNX Steering Committee Update introduction-sc-presentation.mp4 SC-presentation-ONNX-Community-Meetup-2023.pdf	**Alexander Eichenberger**, IBM **Andreas Fehlner**, TRUMPF Laser **Mayank Kaushik**, NVIDIA **Prasanth Pulavarthi**, Microsoft **Saurabh Tangri**, Intel
9:15am: Community Presentations (120 mins)	
The New ONNX Model Zoo mlagility.mp4 01_Groq-ONNX Meetup presentation - Ramakrishnan Sivakumar.pdf *ONNX (Open Neural Network Exchange) Model Zoo has been an invaluable resource for AI and machine learning practitioners. However, we believe there is immense untapped potential that remains to be explored. Recognizing the need to cater to an evolving technological landscape and the diverse needs of users, we embarked on a mission to revamp the ONNX Model Zoo.* *Our objective has been twofold: Firstly, to augment the user experience by introducing a more intuitive, user-friendly interface, and efficient navigation features, making it easier for users to find the models that meet their requirements. Secondly, we strive to broaden the spectrum of models offered in the repository. We are expanding the selection of models to include cutting-edge architectures from various domains such as Computer Vision, Natural Language Processing, Audio, Reinforcement Learning, MultiModal, Generative AI and Graph Machine Learning.*	**Ramakrishnan Sivakumar**, Groq *Ramakrishnan Sivakumar is a Machine Learning (ML) Software Engineer at Groq. He specializes in Graph Neural Networks and is proficient in developing end-to-end ML solutions for a variety of complex problems. He works closely with customers and scientists across diverse industries, leveraging his extensive knowledge and experience to help them navigate the intricacies of ML and deep learning.*

Analysis of Failures and Risks in Deep Learning Model Converters: A Case Study in the ONNX Ecosystem *purdue_exporters.mp4* *02_pu-ONNX Day Presentation - Jajal-Davis.pdf* *Software engineers develop, fine-tune, and deploy deep learning (DL) models. They use and re-use models in a variety of development frameworks and deploy them on a range of runtime environments. In this diverse ecosystem, engineers use DL model converters to move models from frameworks to runtime environments. However, errors in converters can compromise model quality and disrupt deployment. The failure frequency and failure modes of DL model converters are unknown.* *In this talk, we present a failure analysis on DL model converters. We characterized failures in model converters associated with ONNX (Open Neural Network eXchange). We analyzed past failures in the ONNX converters in two major DL frameworks, PyTorch and TensorFlow. The symptoms, causes, and locations of failures (for N=200 issues), and trends over time are also reported. We also evaluated present-day failures by converting 8,797 models, both real-world and synthetically generated instances. The consistent result from both parts of the study is that DL model converters commonly fail by producing models that exhibit incorrect behavior: 33% of past failures and 8% of converted models fell into this category. Our results motivate work to make DL software simpler to maintain, extend, and validate.*	**James C. Davis**, Purdue *James C. Davis is a professor of electrical & computer engineering at Purdue University. His research is in software engineering, studying the engineering of software-intensive computing systems. He is specifically interested in how software-intensive systems fail and how these failure modes can be mitigated.* **Purvish Jajal**, Purdue *Purvish Jajal is a PhD student at Purdue University where he works in pre-trained models and efficient machine learning.*
Olive: a user-friendly toolchain for hardware-aware model optimization olive.mp4 03_MS-Olive_ONNX Meetup.pdf *Olive is an easy-to-use hardware-aware model optimization tool that composes industry-leading techniques across model compression, optimization, and compilation. Given a model and targeted hardware, Olive composes the best suitable optimization techniques to output the most efficient model(s) for inferring on cloud or edge, while taking a set of constraints such as accuracy and latency into consideration. It works with ONNX Runtime, a high-performance inference engine, as an end-to-end inference optimization solution.*	**Emma Ning**, Microsoft *Emma Ning is a Principal PM in the Microsoft AI Framework team, focusing on AI model operationalization and acceleration with ONNX Runtime/Olive for open and interoperable AI. She is passionate about bringing AI solutions to solve business problems as well as enhancing product experience.*
Extensions to ONNX for Multi-Device Support *multi-gpu.mp4* *04_NVDA-ONNX multi-gpu.pdf* *The AI industry has grown model size at a faster rate than the growth in hardware capabilities. This model growth has made it difficult to run SOTA models utilizing ONNX due to no support for expressing execution of the network on multiple devices. This proposal introduces a few extensions to ONNX that allow expression of multi-device capability with minor changes relative to a single device network. By expressing some optional information to an ONNX network, the model can be scaled to multiple devices.*	**Micah Villmow**, NVIDIA *Micah Villmow is a principal engineer at NVIDIA. He graduated from Florida State University with a bachelors and master's degree in computer science, and started his career as an intern for ATI Graphics, where he implemented the HMAX object recognition model on R5XX GPUs using CTM, CAL, OpenGL, and DirectX, achieving state-of-the-art accuracy at orders of magnitude higher performance over published results. He also worked at AMD, where he was involved in writing high performing GEMM routines with CAL, samples and SDK development with Brook+, and the original OpenCL GPU code generator for Apple and AMD's OpenCL software stack. He then went to work at the stealth chip startup, Soft Machines Inc., as the compiler and performance architect, pushing the boundary for performance on the VISC prototype chips. He has worked on all TensorRT releases since he joined NVidia in 2016.*
On-Device Training with ONNX Runtime *on-device-training.mp4* *05_ms-ONNX Meetup on-device 2023.pdf* *We will be introducing On-Device Training, a new capability in ONNX Runtime (ORT) which enables training models on edge devices without the data ever leaving the device. The new On-Device Training capability extends the ORT-Mobile inference offering to enable training on the edge devices. The goal is to make it easy for developers to take an inference model and train it locally on-device—with data present on-device—to provide an improved user experience for end customers. We will be giving a brief overview of how to enable your applications to use on-device training.*	**Kshama Pawar**, Microsoft *Kshama Pawar is a Program Manager in the AI Platform team at Microsoft. She works on large language model training with ORT, Azure Containers for PyTorch and On-Device training with ORT. She is always looking for ways to make training faster and efficient for developers.* **Baiju Meswani**, Microsoft *Baiju is a software engineer in Microsoft's AI Frameworks group and has been contributing towards extending ONNX Runtime's training capabilities. Most recently, he has been working towards enhancing ONNX Runtime to support training ONNX models on edge devices.*

ONNX Script (pre-recorded) onnx-script.mov 06_MS_ONNX_Script.pdf *ONNX Script enables authoring ONNX using (a subset of) Python. It combines (i) a static translator from a subset of Python into ONNX, (ii) a converter back from ONNX into Python, and (iii) a runtime shim that allows ONNX Script code to be evaluated using an ONNX backend, allowing convenient testing and debugging.*	**G. Ramalingam,** Microsoft *Rama works on ONNX, ONNX Script, and ONNX Runtime. He is a researcher, with interests in programming languages, programming tools, static program analysis, and compilers.*
INT8 Quantization for Large Language Models with Intel Neural Compressor (pre-recorded) neural-compressor.mp4 *The explosive growth of large language models (LLMs) has facilitated a significant number of breakthroughs in fields like text analysis, language translation, and chatbot technologies. However, the deployment of LLMs presents a formidable challenge due to their large parameter (e.g., over 700GB memory required to run BLOOM-176B model in FP32), making them impractical to run on commodity hardware. Users, therefore, have an ongoing demand for methods of compressing LLMs that maintain comparable accuracy while reducing their memory footprint, for which general quantization recipes may not work.* *To compress LLMs with reasonable accuracy, Intel® Neural Compressor integrates as well as enhances SmoothQuant algorithm, which effectively addresses the compression challenge by efficiently compensating for the accuracy loss introduced by activation quantization. Our team has validated the efficacy of this solution on numerous LLMs such as GPT-J, LLaMA, and BLOOM, achieving promising latency on Intel hardware. Furthermore, Intel® Neural Compressor eliminates the gap that exists in exporting int8 PyTorch models to ONNX format, making it ideal for production deployment.* *We continue to upload ONNX models to the ONNX model zoo and Hugging Face hub (e.g., GPT-J and Whisper-large), which can make contributions to the ONNX community.*	**Mengni Wang**, Intel *Mengni is a Software Engineer at Intel focusing on Intel(R) Neural Compressor development and model quantization. She holds a Master's degree from Shanghai Jiao Tong University, she has been with Intel for the past 2 years.*
Editing and optimizing ONNX models with DL Designer (pre-recorded) *dl-designer.mp4* *DL Designer is a GUI-based application with a rich set of features that allow users to edit an ONNX model in a visual way. Its UI also provides users with convenient access to model optimization and model linting features of the NVIDIA Polygraphy and GraphSurgeon libraries.*	**Gaoyan Xie,** NVIDIA Gaoyan Xie is a senior manager of software engineering at NVIDIA. He leads a team building software tools for developing high-performance deep neural networks.
Dynamic Dimension Analysis in onnx-mlir Compiler (pre-recorded) onnx-mlir.mp4 onnx-mlir-dynamic-dimension-analysis.pdf *ONNX models often have dynamic dimensions such as batch size or sequence length, which makes a compiler difficult to generate optimal code or to decide whether an operator is suitable for running on AI accelerators or not. While ONNX provides shape inference to infer the relationship among dynamic dimensions, it is limited to ONNX operators. Meanwhile, MLIR-based compilers often have many intermediate representation levels including operators rather than ONNX operators. In this talk, we introduce a dynamic dimension analysis in onnx-mlir compiler, which utilizes operator's ShapeHelper to analyze the relationship between dynamic dimensions such as whether two dynamic dimensions are equal or not. Every operator in onnx-mlir has a ShapeHelper interface, which makes the analysis applicable to a broad range of models. We also show how the analysis helps the compiler optimize CPU operators or offload them to AI accelerators.*	**Tung D. Le,** IBM *Tung D. Le is a Staff Research Scientist at IBM Research - Tokyo. He has been interested in systematic methods to optimize deep learning frameworks. He is mainly contributing to onnx-mlir, a MLIR-based compiler, that translates ONNX models into native code for different platforms. He had experience in optimizing popular deep learning frameworks such as Caffer, Chainer, and TensorFlow for IBM's machines.* **Alexander Eichenberger,** IBM *Alexandre E. Eichenberger is a Principal Research Staff Member at IBM T.J. Watson Research Center, Yorktown Heights, NY. He is active in compiler technology for high performance, most recently in lowering ONNX AI models to the latest IBM Z machine with integrated AI Accelerator. He leads the ONNX-MLIR effort, an open-source effort that includes several leading AI companies and is on the ONNX Steering Committee. Prior interests included SIMD code generation and multi-threading support on IBM's supercomputers.* **Tong Chen,** IBM *Tong Chen is a research staff member at IBM T.J. Watson Research Center. He works in compiler optimization, machine learning model optimization and program security.*

HE-MAN – Homomorphically Encrypted MAchine learning with oNnx models (pre-recorded), mci-he-man.mp4 10_mci_HEMAN.pdf *In this talk, I will present our latest publication: HE-MAN – Homomorphically Encrypted MAchine learning with oNnx models.* *Machine learning (ML) algorithms play a crucial role in the success of products and services, especially with the abundance of data available. Fully homomorphic encryption (FHE) is a promising technique that enables individuals to use ML services without sacrificing privacy. However, integrating FHE into ML applications remains challenging. Existing implementations lack easy integration with ML frameworks and often support only specific models.* *To address these challenges, we present HE-MAN, an open-source two-party machine learning toolset. HE-MAN facilitates privacy-preserving inference with ONNX models and homomorphically encrypted data. With HE-MAN, both the model and input data remain undisclosed. Notably, HE-MAN offers seamless support for a wide range of ML models in the ONNX format out of the box. We evaluate the performance of HE-MAN on various network architectures and provide accuracy and latency metrics for homomorphically encrypted inference.*	**Martin Nocker,** MCI *2013-2016 Bachelor's degree in Computer Science from University of Innsbruck, Austria, 2016-2018 Master's degree in Electrical and Computer Engineering from Technical University of Munich (TUM), Germany. 2019-2020 Software Engineer at Swarovski, Wattens, Austria* *2021-now Research Assistant at MCI (Management Center Innsbruck), Austria* *2021-now PhD Student at University of Rostock, Germany*
11:15am: Break (10 mins)	
11:25am: ONNX SIGs and WGs Updates (50 minutes)	
Architecture/Infrastructure SIG Update sig-arch-infra.mp4 sig-infra.pdf	**Liqun Fu,** Microsoft **Ke Zhang**, Ant
Converters SIG Update sig-convertors.mp4 sig-convertors.pdf	**Thiago Crepaldi**, Microsoft **Kevin Chen**, NVIDIA
Model Zoo / Tutorials SIG Update sig-model-zoo.mp4 sig-model-zoo.pdf	**Jacky Chen**, Microsoft
Compiler SIG Update sig-compilers.mp4 sig-compiler.pdf	**Alexander Eichenberger,** IBM **Philip Lassen**, Groq
Operators SIG Update (pre-recorded) sig-operators.mp4	**Michal Karzynski,** Intel **Ganesan Ramalingam,** Microsoft
Pre-processing WG Update (pre-recorded) sig-preprocessing.mp4 sig-preprocessing.pdf	**Joaquin Anton**, NVIDIA
12:15pm: Break (10 mins)	
12:25pm: Roundtable discussions (25 mins)	
Trusted AI roundtables-trusted-ai.mp4	**Saurabh Tangri**, Intel **Rodolfo (Gabe) Esteves**, Intel

ONNX 2.0 Ideas roundtables-onnx-2.mp4	**Prasanth Pulavarthi**, Microsoft
12:50pm: Conclusion	